基于差分进化的改进狼群算法研究 *

王盈祥 [1]，陈民铀 [1]，程庭莉 [1]，盛　琪 [1]，董龙昌 [1]，李　哲 [2]

(1. 重庆大学 电气工程学院 输配电装备及系统安全与新技术国家重点实验室, 重庆 400044; 2. 国网重庆市电力公司电力科学研究院, 重庆 400044)

摘　要：针对传统狼群算法（WPA）存在易陷入局部最优解、计算资源耗费大、鲁棒性低等问题，提出一种基于差分进化的改进狼群算法（DWPA）。首先，通过引入探狼搜索因子、猛狼最大奔袭次数、自适应围攻步长、差分进化策略等对传统狼群算法进行了改进，降低算法计算耗费的同时提高了算法的全局搜索能力；然后，运用马尔可夫链理论证明了 DWPA 的收敛性；最后，对 13 个测试函数进行寻优测试并与 WPA 等 4 种算法进行对比分析。测试结果表明，DWPA 具有良好的鲁棒性和全局搜索能力，在求解多峰、高维、不可分函数方面的寻优能力尤为突出。

关键词：狼群算法；局部最优解；鲁棒性；差分进化；马尔可夫链

中图分类号：TP301.6　　doi: 10.3969/j.issn.1001-3695.2018.02.0083

Research of improved wolf pack algorithm based on differential evolution

Wang Yingxiang[1], Chen Minyou[1], Cheng Tingli[1], Sheng Qi[1], Dong Longchang[1], Li Zhe[2]

(1. *State Key Laboratory of Power Transmission Equipment & System Security*, *School of Electrical Engineering*, *Chongqing University*, *Chongqing* 400044, *China*; 2. *Chongqing Electric Power Company Electric Power Research Institute of State Grid*, *Chongqing* 400044, *China*)

Abstract: Aiming at the problems of traditional wolf pack algorithm (WPA) : easy to fall in to local optimal, large computational resource cost and low robustness, propose an improved wolf pack algorithm based on differential evolution (DWPA) . First of all, propose search wolf search factor, maximum number of raid wolves, adaptive siege step size and differential evolution strategy to improve the traditional wolf pack algorithm, which can not only reduce the computational cost of the algorithm but also improve the global search ability. Then, prove the convergence of DWPA applying the Markov process. Finally, conduct optimization test on 13 functions and then compare it with WPA and other 4 algorithms. The test results show that DWPA has great robustness and global search ability, especially has an excellent optimizing ability in multi-peak, high-dimension, indivisible functions.

Key words: wolf pack algorithm; local optimal; robustness; differential evolution; Markov process

0　引言

群集算法主要是通过模拟生物进化和生物种群行为来求解优化问题的智能算法[1]。常见的群集智能算法主要有粒子群算法（PSO）[2~4]、鱼群算法（FSA）[5,6]、蚁群算法（ACO）[7]、差分进化算法(DE)[8]等，它们在解决优化问题时各有千秋，为某些复杂问题的解决提供了良好的思路。

狼群算法最早是由 Yang 等人[9]提出，2013 年，吴虎胜等人[10]在分析狼群分工协作围捕猎物行为的基础上，提出了一种全新的狼群算法（WPA）。算法虽问世时间较短，但由于其性能较好，已被广泛应用于无人机航迹规划[11]、水电站水库优化调度[12]等人类生产活动中。WPA 通过模拟狼群分工协作捕猎的特征，抽象出游走、召唤、奔袭和围攻等行为与“胜者为王”的头狼产生机制和“强者生存”的群体更新机制来进行优化问题求解，具有较好的寻优性能。但也不可避免地存在一些不足之处：易陷入局部最优解、计算资源耗费大、鲁棒性低等。文献[13]提出一种改进的狼群算法，该算法根据传统狼群算法基本思想提出探狼更新规则并引入相位因子，同时优化了传统狼群算法步长的种类，设计了新的猛狼位置更新公式，通过测试函数仿真模拟验证了该算法的有效性。文献[14]结合文化算法提

收稿日期：2018-02-05；**修回日期：**2018-03-26　**基金项目：**国家自然科学基金资助项目（51177177，61105125，51477019）；国家自然国家电网公司科技项目（5220001600V6）

作者简介：王盈祥（1993-），男，江西赣州人，硕士研究生，主要研究方向为主动配电网经济调度（jxgzwyx@126.com）；陈民铀（1954-），男，教授，博士，主要研究方向为人工智能、电力系统控制与优化；程庭莉（1988-），女，博士研究生，主要研究方向为多目标优化，电力系统优化运行；盛琪（1992-），女，黑龙江哈尔滨人，硕士研究生，主要研究方向为规模化电动汽车接入对配电网谐波的影响分析；董龙昌（1993-），男，河南洛阳人，硕士研究生，主要研究方向为配电网中车网协调优化控制；李哲（1989-），男，河南周口人，工程师，硕士，主要研究方向为智能配电网、微电网技术.

出了一种文化狼群算法，该算法可有效解决人工狼搜索的盲目性问题，通过对三个复杂函数的测试分析，验证了该文化狼群算法的有效性。文献[15]将 PSO 算法中求解当前局部最优的思想引入到狼群算法的游走和召唤行为中，并利用混沌法对得到的次优解进行优化。改进后的算法大大提高了搜索的准确度并避免了陷入局部极值，并通过仿真验证了该算法的有效性。

针对 WPA 存在的不足之处，本文提出一种基于差分进化的改进狼群算法（DWPA）。该算法在对传统 WPA 算法智能行为改进的基础上，引入差分进化中的交叉、变异、选择等操作，大大提高了算法的寻优性能及鲁棒性。

1　DWPA 算法原理

DWPA 算法继承了 WPA 算法的所有行为并对其进行了改进。主要改进如下：a)WPA 中游走行为缺乏引导，一旦游走方向数 h 确定，游走方向便随之确定，无法全面搜索解空间，易使算法陷入局部最优，故在 DWPA 游走行为中引入探狼搜索因子，使得算法寻优能遍历整个空间;b)WPA 中的奔袭行为需要分析计算人工狼与头狼之间的距离并要求所有人工狼与头狼之间距离小于临界距离时才结束奔袭行为，在面对高维函数的寻优时，必然造成大量计算资源的耗费，同时由于临界距离这一唯一行为终止条件的存在将可能使得算法无法跳出奔袭行为，降低算法的鲁棒性，故在 DWPA 奔袭行为中除去人工狼距离计算过程，并引入猛狼最大奔袭次数和头狼替换两个条件作为行为终止条件，防止算法陷入无限循环并提高计算效率和算法鲁棒性;c)WPA 中围攻行为不具有自适应特性，决策变量取值范围和步长因子一旦确定，所有人工狼将以同一确定步长进行围攻，故在 DWPA 围攻行为中引入自适应围攻步长，使得距离头狼相对较远的人工狼以相对较大的步长向头狼靠近，距离头狼相对较近的人工狼以较小步长对头狼附近区域进行更加精细的搜索，兼顾了算法的局部搜索和全局搜索能力。同时，针对 WPA 算法本身易陷入局部最优的问题，结合差分进化算法在局部寻优方面的优势[8]，在改进 WPA 算法中加入差分进化的变异、交叉和选择操作，使算法能跳出局部最优，提高算法全局搜索能力。

综上，DWPA 算法由游走、召唤、奔袭、围攻、变异、交叉、选择等七种行为及“胜者为王”的头狼产生、“强者生存”的群体更新等两种机制组成。

下面以最小值寻优问题为例，详细介绍 DWPA 算法的主要过程：

1）狼群初始化

采用式（1）进行狼群初始化，产生 n 匹人工狼：

$$X_i^t = X_i^L + rand \times \left(X_i^U - X_i^L\right), i = 1,2,\cdots,n \qquad (1)$$

其中：X_i^t 表示第 t 代种群中第 i 匹人工狼；X^L 和 X^U 分别为变量 $X = \left(x_1, x_2, \cdots, x_D\right)$ 取值的下界和上界，D 为变量维数；$rand \in (0,1)$ 内的随机数。

2）头狼产生规则

在初始化产生的种群中，通过式（2）计算每一匹人工狼的适应度值，适应度值越小的位置视为猎物浓度越大，距离所求最优解越近，选取具有最优适应度值的人工狼作为本次迭代过程中的头狼，若有多匹人工狼适应度值都是最优值，则任选一匹作为头狼，头狼适应度值为 Q_{lead}；在进化过程中，将每次进化后具有最优适应度值的人工狼与前一代中的头狼进行比较，若当前人工狼的适应度值更优，则该人工狼替代头狼成为新的头狼。对于最小值问题，当 $Q_i < Q_{lead}$ 时，视为 Q_i 优于 Q_{lead}。

$$Q = f\left(X\right) \qquad (2)$$

其中：$f(X)$ 为所求最小值问题的目标函数。

3）游走行为

除头狼外，在当前猎物群中选取适应度值最优的 S_{num} 匹人工狼作为探狼，在空间中搜寻猎物。S_{num} 取值为 $\left[n/(\alpha+1), n/\alpha\right]$ 间的整数，其中，α 为探狼比例因子。探狼朝向 h 个方向分别前进一步并记录每前进一步后所处位置的适应度值，然后退回原位置，则向第 p(p=1,2,...,h)个方向前进后，探狼 i 在第 d(d=1,2,...,D)维空间中所处的位置如式（3）所示。

$$x_{id}^P = x_{id} + \gamma \times step_a^d \qquad (3)$$

其中：$step_a$ 为探狼游走步长，γ 为探狼搜索因子，为$[-1,1]$内的随机数。

此时，探狼所处位置的适应度值为 Q_{ip}，选取适应度值最小且小于当前位置适应度值 Q_{io} 的方向前进一步，更新探狼信息 X_i。将一次游走行为结束后的具有最小适应度值 Q_{min} 的人工狼与头狼 Q_{lead} 进行比较，若 $Q_{min}<Q_{lead}$，则 $Q_{lead}=Q_{min}$，适应度值最小的探狼替代原头狼成为新的头狼，并发起召唤行为;否则，重复游走行为，直到某匹狼 i 所处的位置适应度值 $Q_i<Q_{lead}$，或游走次数 T_1 达到最大游走次数 T_{1max}。需要指出的是，每匹探狼由于适应度值存在差异，其搜索方式也存在差异，故每匹探狼 h 的取值是不同的，将在$[h_{min},h_{max}]$内取随机整数，一般来说 h 越大，探狼搜索越精细但也将影响算法运行速率。

4）奔袭行为

头狼通过嚎叫发起召唤行为，召集附近猛狼快速向头狼靠近。猛狼 i 在第 k+1 次进化时，在第 d(d=1,2,...,D)维空间中所处的位置为如式（4）所示。

$$x_{id}^{k+1} = x_{id}^k + step_b^d \times \left(g_d^k - x_{id}^k\right) / \left|g_d^k - x_{id}^k\right| \qquad (4)$$

其中：$step_b$ 为猛狼奔袭步长，g_d^k 为第 k 代群体头狼在第 d(d=1,2,...,D)维空间中的位置。

在猛狼奔袭的过程中，将每次奔袭结束后具有最小适应度值 Q_{min} 的猛狼与头狼 Q_{lead} 进行比较，若 $Q_{min}<Q_{lead}$，则 $Q_{lead}=Q_{min}$，该猛狼成为新的头狼，奔袭行为结束，进入下一行为；否则继

续奔袭直到某匹猛狼 i 的适应度值小于头狼适应度值或奔袭达到最大奔袭次数 T_{2max}。

5）围攻行为

猛狼奔袭行为结束后，将联合所有的探狼，以头狼所在的位置为猎物移动位置，执行围攻行为。对于第 k 代狼群，假定头狼在第 d (d=1,2,...,D)维空间中的位置为 G_d^k，则狼群的围攻行为如式（5）所示。

$$x_{id} = x_{id}^k + \lambda \times step_i^d \times (G_d^k - x_{id}^k) / \left|G_d^k - x_{id}^k\right| \quad (5)$$

其中：λ 为[-1,1]间均匀分布的随机数；$step_i$ 为人工狼 i 执行围攻行为时的自适应围攻步长，距离头狼相对较远的人工狼以相对较大的步长向头狼靠近，距离头狼相对较近的人工狼以较小步长对头狼附近区域进行更加精细的搜索。以搜索空间上下界和搜索空间中的适应度最差值与最优值的差作为基准值，以自适应围攻步长与人工狼适应度值和头狼适应度值的差成正相关关系来建立自适应围攻步长 $step_i$ 如式（6）所示。

$$step_i = (Q_i - Q_{lead}) \times \frac{X^U - X^L}{\max Q_0 - \min Q_0} \quad (6)$$

其中：$\max Q_0$ 和 $\min Q_0$ 分别为算法初始化产生的人工狼的最差适应度值和最优适应度值。

若围攻后的人工狼所处位置的适应度值小于原位置的适应度值，则更新人工狼位置信息，否则，人工狼位置不变。

特别地：游走步长 $step_a$，奔袭步长 $step_b$ 在待寻优空间中存在如式（7）所示关系：

$$step_a = step_b / 2 = \left|X^U - X^L\right| / S \quad (7)$$

其中：S 为步长因子，表示人工狼在待寻优空间中搜索的精细程度。

6）变异行为

在执行围攻行为后的群体中，随机选择两匹不同的人工狼（第 k 代）X_1^k、X_2^k，将两者的矢量差赋予权值后与第三匹随机选择的人工狼 X_3^k 相加，得到人工狼变异矢量 v_i^{k+1}，对于任意人工狼目标矢量 X_i^k 有

$$v_i^{k+1} = X_3^k + F * \left(X_1^k - X_2^k\right) \quad (8)$$

其中：F 为缩放因子，一般在[0,2]中取值，表示对矢量差的缩放程度。

7）交叉行为

狼群经过变异行为后，对于得到的人工狼变异矢量 v_i^{k+1} 和人工狼目标矢量 X_i^k，按式（9）进行参数混合，生成人工狼试验矢量 u_i^{k+1}：

$$u_{ij}^{k+1} = \begin{cases} v_{ij}^{k+1} & rand(j) \le CR or j = randn(i) \\ x_{ij}^k & rand(j) > CR or j \ne randn(i) \end{cases} \quad (9)$$

其中：j 表示人工狼的第 j 个变量，$randn(i) \in [1,2,\cdots,D]$，为随机选择的维数变量的标识；$CR$ 为交叉概率因子，一般在[0,1]中取值，CR 越大越有利于算法进行局部搜索和加快收敛速度，CR 越小越有利于保持狼群多样性和全局搜索。

8）选择行为

狼群经过交叉行为后，对于得到的人工狼试验矢量 u_i^{k+1} 与人工狼目标矢量 X_i^k，若试验矢量的适应度值小于目标矢量，则用 u_i^{k+1} 取代 X_i^k 成为第 k+1 代并更新人工狼位置，否则，直接用 X_i^k 作为第 k+1 代更新，即式（10）。

$$X_i^{k+1} = \begin{cases} u_i^{k+1} & Q(u_i^{k+1}) < Q(x_i^k) \\ X_i^k & Q(u_i^{k+1}) \ge Q(x_i^k) \end{cases} \quad (10)$$

对于得到的人工狼矢量 X_i^{k+1}，找出其中具有最小适应度值 Q_{min} 的人工狼并与头狼适应度 Q_{lead} 进行比较，若 $Q_{min}<Q_{lead}$，则 $Q_{lead}=Q_{min}$，适应度值最小的人工狼选为头狼。

9）狼群更新机制

狼群捕食得到的猎物，并不是平均分配给每一匹狼，将按照“肉弱强食”的原则进行食物分配，这就可能导致弱小的狼被饿死，这一原则通过在算法中去掉适应度值最差的 R 匹狼，同时按式（1）产生 R 匹人工狼来进行模拟。R 为 $[n/(2\times\beta), n/\beta]$ 之间的随机整数，β 为狼群更新比例因子。

通过以上行为，使得狼群不断进行更新进化，直到算法达到限定的最大迭代次数 G 或算法获得的最优值达到预设精度时结束，故有 DWPA 求解优化问题流程图如图 1 所示。

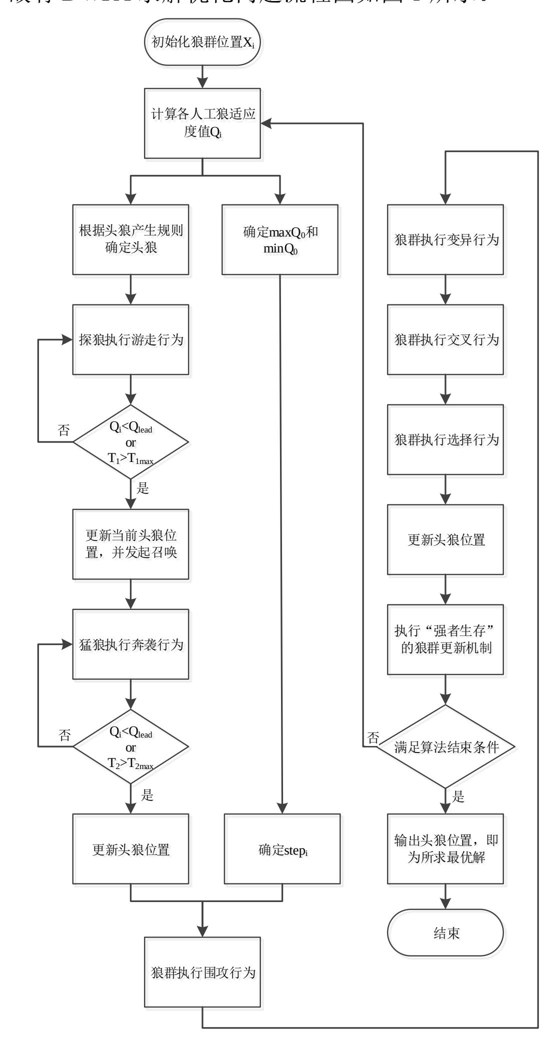


图 1　DWPA 求解优化问题流程图

2　DWPA 收敛性分析

马尔可夫（Markov）链是一种无后效性的随机过程，常被用于证明算法的收敛性[16]。DWPA 是不断重复游走、召唤、奔袭、围攻、交叉、变异、选择和狼群更新行为的过程，种群的每种行为都只与当前群体状态有关而与之前的状态无关，由此可知，DWPA 种群序列符合 Markov 链。

引理 1　文献[17]中证明了若某智能算法满足如下两个条件:a)对解空间中任意两解 x_1 和 x_2， x_2 是 x_1 由算法中的各种算子可达的；b)种群序列 $Q_1,Q_2,\cdots,Q_N$ 是单调的。则该智能算法以概率 1 收敛于问题的全局最优解。

通过以上结论可知，如果 DWPA 能够满足引理 1 的两个条件，则说明 DWPA 是以概率 1 收敛问题的全局最优解的。

参考引理 1 的结论，对于 DWPA 收敛性问题，若其满足以下两个条件：a)DWPA 种群序列的 Markov 链是遍历的；b)DWPA 的序列解是一个有限齐次 Markov 链。则 DWPA 将以概率 1 收敛于问题的全局最优解。

假设 DWPA 搜索的解空间为 Ω，游走、奔袭、围攻、交叉、变异、选择和狼群更新行为所引起的状态空间的状态转移分别用转移矩阵 Y、B、W、J、V、X 和 G 表示，则 DWPA 种群的 Markov 链的转移矩阵定义为

$$P = Y \times B \times W \times J \times V \times X \times G \qquad (11)$$

在证明 DWPA 的收敛性之前，首先给出以下几个定义：

定义 1　假设 P_{ij} 为 Markov 链的转移概率矩阵，若有对于 $\forall i,j \in \Omega$，$\exists k \ge 1$，使得 $P_{ij}^k > 0$，则称该 Markov 链是不可约的。

定义 2　假设有非空集合 $I = \{k \mid k \ge 1, P_{ij}^k > 0, \forall i,j \in \Omega\}$，若该集合 I 的最大公约数为 1，则称该 Markov 链是非周期的。

定义 3　假设 $\mu_i = \sum_k^{\infty} kP_{ij}^k$，对于常返状态 i，若有 $\mu_i < +\infty$，则称 μ_i 为正常返的。特别地，当 i 为正常返的且非周期的，则该 Markov 链是遍历的。

下面证明 DWPA 收敛性：

推理 1　DWPA 种群序列的 Markov 链是遍历的。

推理 1 的证明，将分以下三个步骤进行：

a)证明 DWPA 的种群序列 Markov 链是不可约的。

假设 $Q_k = \{X_1, X_2, \cdots, X_N\}$ 为算法的第 k 代种群， X_i 为第 i 匹人工狼的状态。由于种群 Markov 链的转移概率矩阵 $P_{ij} = P\{Q_{k+1} = j \mid Q_k = i, k \ge 1\}$ 只与始末状态 i,j 相关，且显然有 $Q_k > 0$ 恒成立，则有种群转移概率矩阵 P 为正定矩阵，由定义 1，DWPA 种群序列的 Markov 链是不可约的。

b)证明 DWPA 种群的 Markov 链是非周期且不可约的。

对于某一给定的 $k > 0$，由①中得到的算法 Markov 链是不可约的可知，必然 $\exists j \in \Omega$，使得 $P_{ij} > 0$ 成立，结合定义 2 可知 $k = 1$。因此可得 I 的最大公约数为 1，故有 DWPA 种群的 Markov 链是非周期的，结合①可知 DWPA 种群的 Markov 链是非周期且不可约的。

c)证明 DWPA 种群的 Markov 链是遍历的。

由于转移矩阵 Y、B、W、J、V、X 和 G 的值均在 [0,1] 内，且 P_{ij} 为状态 i 经过各种行为转化为状态 j 的概率，必有 $0 \le P_{ij} \le 1$，设 $\sigma = \max\{P_{ij} \mid \forall i,j \in \Omega\}$，则由 Canchy-Riemann 方程及定义 3 可知： $\mu_i = \sum_k^{\infty} kP_{ij}^k \le \sum_k^{\infty} k\sigma^k < \infty$。

综上分析可知，DWPA 种群序列的 Markov 链是遍历的，推理 1 得证。

推理 2　DWPA 得序列解是一个有限齐次 Markov 链

证明　由于每一代种群 Q_k 都是有限的，故其 Markov 链必是有限的；其次，算法不断重复游走、召唤、奔袭、围攻、交叉、变异、选择和狼群更新行为的过程寻找更加优质的猎物，个体的更新都具有优质选择的特性。群体第 k+1 代的产生仅与第 k 代有关，而与 P_{ij} 和 k 无关。DWPA 经过反复迭代寻优，必能得到一组序列解，且该序列是一个有限齐次 Markov 链，推理 2 得证。

综上推理 1 和 2 可知，DWPA 种群序列的 Markov 链是遍历的且其序列解是一个有限齐次 Markov 链，至此证明 DWPA 以概率 1 收敛于所求问题的全局最优解。

3　DWPA 算法仿真测试分析

3.1　基本测试函数介绍

为了测试 DWPA 算法的性能及其有效性，本文选用 13 个数学特征和规模不同的标准单目标测试函数对其进行测试，各测试函数如表 1 所示；在表 1 特征栏中，“U”表示单峰函数，即该函数在定义域内只有一个极值，即全局最优值，没有局部最优值；“M”表示多峰函数，即该函数在定义域内有多个局部极值，主要用来检测算法的全局寻优和避免早熟的能力[18]；“S”表示可分函数，“N”表示不可分函数，若某具有 n 个变量的函数能用这 n 个单变量函数之和来表示该函数，则该函数为可分函数，否则，为不可分函数。不可分函数相对可分函数将更加复杂，在寻优过程中对算法的性能要求更高。在表 1 的定义域栏中，集合部分为变量取值范围，上标为函数的维数 D，测试函数维数 D 越高，算法的运算复杂度越大，对算法性能的要求也越高。

3.2　算法对比验证及分析

为了测试 DWPA 算法的性能，本文同时用 DWPA、WPA、

PSO、DE 和 FSA 算法对以上 13 个函数分别重复进行 100 次寻优测试并记录寻优的最优值、最差值、平均值、标准差和计算成功率五个方面进行对比分析。实验环境为：Intel(R) Core^{TM} i5-4590 CPU @3.30GHz，内存 16 GB，Win7 专业版 64 位操作系统，MATLAB R2013b。

表 1　测试函数

函数名	函数表达式	特征	定义域	全局最优
Sphere	$f_1=\sum_{i=1}^{D}x_i^2$	US	$[-10,10]^2$	0
Matyas	$f_2=0.26\left(x_1^2+x_2^2\right)-0.48x_1x_2$	UN	$[-10,10]^2$	0
Booth	$f_3=\left(x_1+2x_2-7\right)^2+\left(2x_1+x_2-5\right)^2$	MS	$[-10,10]^2$	0
Eggcrate	$f_4=x_1^2+x_2^2+25(\sin^2x_1+\sin^2x_2)$	MN	$[-\pi,\pi]^2$	0
Trid6	$f_5=\sum_{i=1}^{D}(x_i-1)^2-\sum_{i=2}^{D}x_ix_{i-1}$	UN	$[-36,36]^6$	-50
Sumsquares-1	$f_6=\sum_{i=1}^{D}ix_i^2$	US	$[-10,10]^{10}$	0
Rastrigin	$f_7=\sum_{i=1}^{D}[x_i^2-10\cos(2\pi x_i)+10]$	MS	$[-10,10]^{60}$	0
Griewank	$f_8=\frac{1}{4000}\sum_{i=1}^{D}x_i^2-\prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})+1$	MN	$[-600,600]^{100}$	0
Quadric	$f_9=\sum_{i=1}^{D}(\sum_{k=1}^{i}x_k)^2$	MS	$[-30,30]^{100}$	0
Sumsquares-2	$f_{10}=\sum_{i=1}^{D}ix_i^2$	US	$[-10,10]^{100}$	0
Rosenbrock	$f_{11}=\sum_{i=1}^{D-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	MN	$[-30,30]^{100}$	0
Ackley-1	$f_{12}=-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2})-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i))+20+e$	MN	$[-32，32]^{100}$	0
Ackley-2	$f_{13}=-20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D}x_i^2})-\exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i))+20+e$	MN	$[-32，32]^{200}$	0

计算成功率定义为成功计算次数占总计算次数的百分比[10]。对于每次寻优计算得到的结果 Q，当 Q 满足式（12）时，表示此次寻优成功。

$$\begin{cases}\left|\frac{Q-Q_{best}}{Q_{best}}\right|<1E-03 & Q_{best}\neq 0\\ \left|Q-Q_{best}\right|<1E-03 & Q_{best}=0\end{cases}\tag{12}$$

其中：Q_{best} 为所求函数的理论全局最优值。

寻优计算的最大迭代次数设为 2000，DWPA、WPA、PSO、DE 和 FSA 算法的种群规模设为 50，其中：WPA 算法的参数依据文献[10]选取，PSO 算法的参数依据文献[2]选取，DE 算法的参数依据文献[8]选取，FSA 算法的参数依据文献[6]选取，整理如表 2 所示；五种算法对以上 13 个函数寻优计算的结果如表 3 所示。

表 2　各算法参数对照表

算法	参数取值
PSO	惯性权重 $\omega=0.7298$，学习因子 $c_1=c_2=2$，个体速度范围[-0.5,0.5]
DE	缩放因子 F=0.6，交叉概率因子 CR=0.9
FSA	最多试探次数 try_{num}=100,感知距离 $\upsilon=1$，拥挤度因子 de=0.618，步长 step=0.1
WPA	探狼比例因子 $\alpha=4$，最大游走次数 T_{max}=20,距离判定因子 $\omega=500$，步长因子 s=1000, 更新比例因子 β=6
DWPA	探狼比例因子 $\alpha=4$，最大游走次数 T_{1max}=10,最大奔袭次数 T_{2max}=10,步长因子 S=1500, 更新比例因子 β=3, 缩放因子 F=0.6，交叉概率因子 CR=0.9

由表 3 中各种算法的寻优计算结果可以看出：

a) 对于单峰、低维的可分函数 Sphere 及不可分函数 Matyas,5 种算法都寻优成功且具有较好的性能，其中 DWPA 和

WPA 的寻优能力相当，尤其在求解不可分函数 Matyas 时，DWPA 和 WPA 性能远高于其他三种算法。

b)对于多峰、低维的可分函数 Booth，5 种算法都寻优成功且性能都较好，其中 PSO 寻优性能最佳，此时 DWPA 性能优势并不明显。

c)对于多峰、低维的不可分函数 Eggcrate,FSA 存在由于陷入局部最优值而使寻优失败的情况，使其寻优成功率较低，其他四种算法全部寻优成功，此时 DWPA 具有最佳寻优性能。

d）对于单峰、不可分函数 Trid6，DWPA 和 PSO 都寻优成功且性能较好，DE 和 WPA 存在陷入局部最优而寻优失败的情况，FSA 寻优完全失效。

e)对于单峰、可分函数 Sumsquares-1，DWPA、WPA、PSO、FSA 都寻优成功，其中 DWPA 和 WPA 具有最佳寻优性能，DE 寻优完全失效。对于高维函数，PSO、DE 和 FSA 都无法跳出局部最优而导致寻优失败，不能获得较好的寻优效果；只有 WPA 和 DWPA 有一定的寻优能力，故以下只对比 WPA 和 DWPA 的寻优性能。

表 3　各算法寻优结果

函数	算法	最优值	最差值	平均值	标准差	计算成功率
Sphere（2 维）	PSO	5.87E-108	2.51E-99	2.35E-100	6.30E-100	100%
	DE	2.37E-23	6.24E-16	5.88E-18	8.38E-16	100%
	FSA	2.40E-16	4.73E-09	8.64E-10	7.53E-09	100%
	WPA	0	0	0	0	100%
	DWPA	0	0	0	0	100%
Matyas（2 维）	PSO	2.33E-13	9.70E-10	1.11E-11	2.01E-11	100%
	DE	1.97E-17	9.77E-15	5.01E-16	2.31E-16	100%
	FSA	8.33E-14	5.78E-11	9.89E-12	1.32E-11	100%
	WPA	3.01E-101	6.18E-94	2.97E-96	1.12E-94	100%
	DWPA	2.23E-101	5.65E-95	3.42E-97	7.20E-96	100%
Booth（2 维）	PSO	0	0	0	0	100%
	DE	6.55E-20	1.87E-17	2.56E-18	8.11E-17	100%
	FSA	1.67E-11	2.43E-09	5.98E-10	6.27E-10	100%
	WPA	7.22E-10	6.46E-07	2.13E-08	1.71E-08	100%
	DWPA	6.15E-11	5.45E-09	1.12E-10	1.33E-09	100%
Eggcrate（2 维）	PSO	2.52E-189	9.47E-182	5.21E-183	8.57E-183	100%
	DE	4.31E-22	6.66E-19	1.85E-19	1.74E-19	100%
	FSA	5.74E-07	9.49E+00	1.52E+00	3.51E+00	5%
	WPA	1.27E-171	1.25E-164	2.65E-165	0	100%
	DWPA	0	0	0	0	100%
Trid6（6 维）	PSO	-50	-50	-50	0	100%
	DE	-50	-49.9387	-49.9969	0.0134	97%
	FSA	-49.9498	-49.9395	-49.9476	0.0041	0
	WPA	-50	-49.9497	-49.9975	0.011	96%
	DWPA	-50	-50	-50	0	100%
Sumsquares-1（10 维）	PSO	2.21E-46	1.16E-34	5.92E-36	2.59E-35	100%
	DE	2.53E-02	1.34E-01	7.58E-02	2.11E-02	0
	FSA	2.13E-12	1.23E-07	2.12E-09	1.87E-09	100%
	WPA	0	0	0	0	100%
	DWPA	0	0	0	0	100%
Rastrigin（60 维）	PSO	1.80E+02	2.68E+02	2.25E+02	2.18E+01	0
	DE	4.80E+02	5.78E+02	5.33E+02	1.99E+01	0
	FSA	1.10E+02	1.22E+03	9.74E+02	5.85E+01	0
	WPA	4.41E-09	3.37E-06	3.75E-07	7.48E-07	100%
	DWPA	0	2.41E-12	1.58E-13	5.20E-13	100%

Griewank（100 维）	PSO	1.79E-01	2.55E-01	2.21E-01	2.06E-02	0
	DE	3.27E-02	9.19E-01	1.01E-01	2.14E-02	0
	FSA	7.83E+03	1.01E+04	9.83E+03	4.54E+02	0
	WPA	3.11E-12	3.88E-01	1.49E-01	9.73E-02	94%
	DWPA	2.44E-15	6.68E-12	1.59E-12	1.95E-12	100%
Quadric（100 维）	PSO	4.37E+02	1.13E+03	7.04E+02	1.50E+02	0
	DE	1.19E+05	2.30E+05	1.83E+05	3.29E+04	0
	FSA	1.31E+05	3.50E+05	2.78E+05	5.56E+04	0
	WPA	5.93E-10	8.64E-07	2.33E-07	3.06E-07	100%
	DWPA	4.15E-12	8.48E-08	1.11E-08	1.89E-08	100%
Sumsquares-2（100 维）	PSO	5.98E+02	1.14E+03	8.57E+02	1.34E+02	0
	DE	5.73E-01	4.72E+03	2.61E+03	1.23E+03	0
	FSA	3.03E+05	4.16E+05	3.61E+05	2.39E+04	0
	WPA	5.02E-09	5.02E-07	1.21E-07	1.37E-07	100%
	DWPA	7.15E-16	6.12E-12	8.78E-13	1.43E-12	100%
Rosenbrock（100 维）	PSO	1.16E+01	1.45E+01	1.29E+01	5.83E-01	0
	DE	5.11E+00	1.58E+01	9.01E+00	2.77E+00	0
	FSA	1.15E+01	1.52E+01	1.31E+01	6.88E-01	0
	WPA	2.59E-11	5.26E-02	3.83E-02	8.48E-03	90%
	DWPA	3.23E-13	2.82E-10	2.39E-11	6.20E-11	100%
Ackley-1（100 维）	PSO	5.04E+00	6.99E+00	6.08E+00	3.83E-01	0
	DE	1.92E+01	2.00E+01	1.98E+01	1.13E-01	0
	FSA	2.13E+01	2.15E+01	2.14E+01	3.42E-02	0
	WPA	7.30E-13	6.36E-02	5.73E-02	1.76E-02	87%
	DWPA	8.88E-16	8.88E-16	8.88E-16	0	100%
Ackley-2（200 维）	PSO	2.71E+00	3.29E+00	3.14E+00	1.42E-01	0
	DE	1.91E+01	1.98E+01	1.96E+01	1.13E-01	0
	FSA	2.13E+01	2.16E+01	2.14E+01	3.45E-02	0
	WPA	1.52E-11	7.17E-01	2.39E-01	4.68E-01	75%
	DWPA	8.88E-16	8.88E-16	8.88E-16	0	100%

f)对于 60 维的多峰、可分函数 Rastrigin，DWPA 和 WPA 寻优成功，DWPA 寻优精度高于 WPA，如图 2 所示。

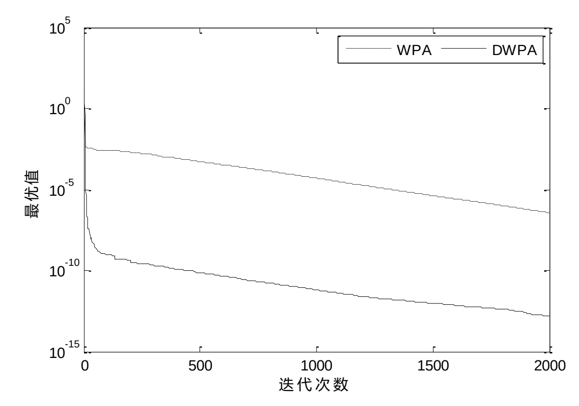


图 2 DWPA 与 WPA 的 Rastrigin 函数平均迭代曲线

g)对于高维、多峰的不可分函数 Griewank 和 Rosenbrock，DWPA 全部寻优成功且性能较稳定，WPA 存在陷入局部最优的情况而使得寻优能力不稳定,如图 3、4 所示。

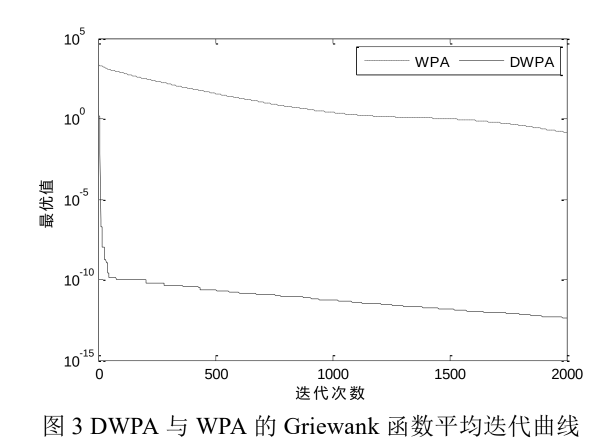


图 3 DWPA 与 WPA 的 Griewank 函数平均迭代曲线

h)对于高维、多峰的可分函数 Quadric 和单峰的可分函数 Sumsquares-2，DWPA 和 WPA 寻优成功，且 DWPA 寻优精度高于 WPA，如图 5 及图 6 所示所示。

i)对于高维的多峰、不可分函数 Ackley，DWPA 寻优性能较稳定，WPA 存在陷入局部最优的情况，且随着 Ackley 函数

维数的增加，其陷入局部最优的可能性增加而导致寻优能力继续下降，如图 7、8 所示。

综上所述，DWPA 对各类单峰/多峰、低维/高维、可分/不可分函数都具有较好的寻优计算能力，与 WPA 相比，DWPA 对于高维、不可分函数的求解不易陷入局部最优值，算法鲁棒性也得到加强。

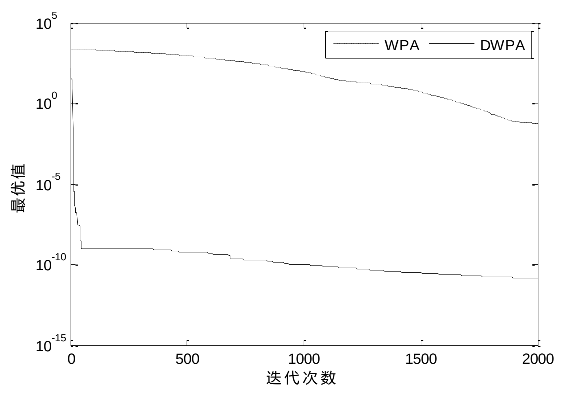


图 4 DWPA 与 WPA 的 Rosenbrock 函数平均迭代曲线

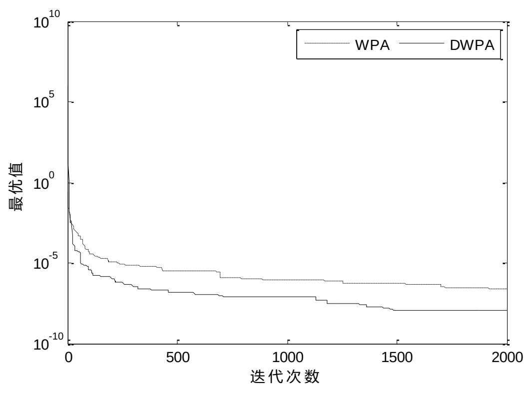


图 5 DWPA 与 WPA 的 Quadric 函数平均迭代曲线

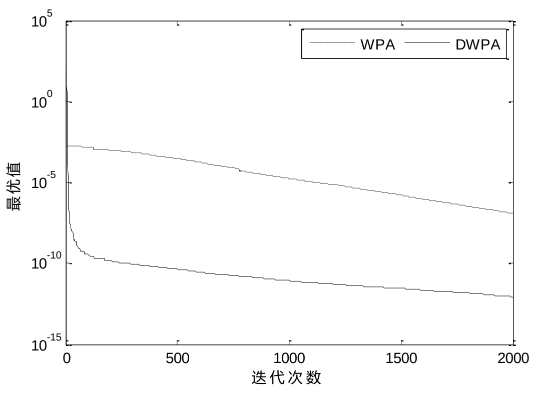


图 6 DWPA 与 WPA 的 Sumsquares-2 函数平均迭代曲线

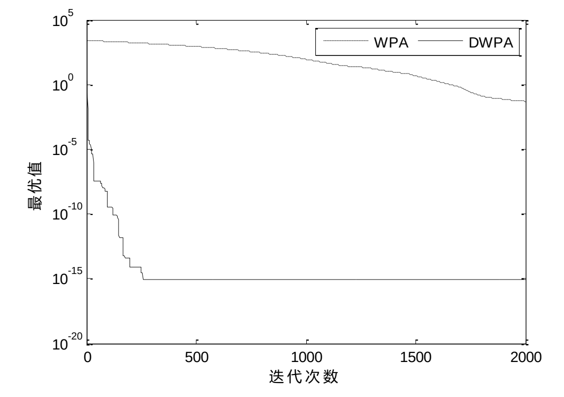


图 7 DWPA 与 WPA 的 Ackley-1 函数平均迭代曲线

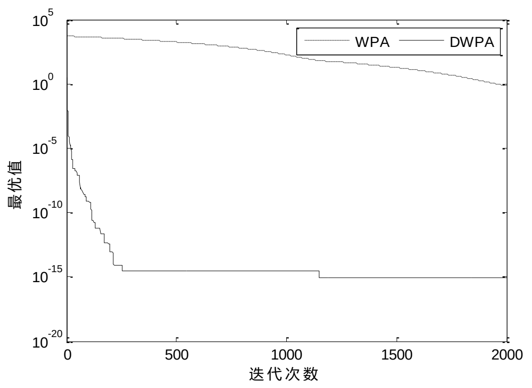


图 8 DWPA 与 WPA 的 Ackley-2 函数平均迭代曲线

4 结束语

为了提高狼群算法的寻优性能，本文提出一种基于差分进化的改进狼群算法（DWPA），在游走行为中引入探狼搜索因子，使得算法寻优能遍历整个空间；在奔袭中除去人工狼距离计算过程，并引入猛狼最大奔袭次数和头狼替换作为两个行为终止条件，防止算法陷入无限循环并提高计算效率和算法鲁棒性；在围攻行为中引入自适应围攻步长，使得距离头狼相对较远的人工狼以相对较大的步长向头狼靠近，距离头狼相对较近的人工狼以较小步长对头狼附近区域进行更加精细的搜索，兼顾算法的局部搜索和全局搜索能力；并在改进 WPA 算法的基础上加入差分进化的变异、交叉和选择操作，使算法能跳出局部最优，提高算法全局搜索能力。通过 13 个测试函数对 DWPA 等 5 种算法进行了性能对比测试。测试结果表明，DWPA 对各类单峰/多峰、低维/高维、可分/不可分函数都具有较好的寻优计算能力，其在求解多峰、高维、不可分函数时能有效跳出局部最优，为高维、不可分等复杂函数的求解提供了一种新的思路和高效的解决方法。但本文只在围攻行为中引入了自适应步长，如何对游走、奔袭行为中的固定步长进行改进以获得更佳的寻优效果需要进一步研究。

参考文献：

[1] 丁建立，陈增强，袁著祉. 智能仿生算法及其网络优化中的应用研究进展 [J]. 计算机工程与应用，2003, 39 (12): 10-15. (Ding Jianli, Chen Zengqiang, Yuan Zhuzhi. Research progress of intelligent bionic algorithm and its application in network optimization [J]. Computer Engineering and Applications, 2003, 39 (12): 10-15.)

[2] Kennedy J, Eberhart R. Particle swarm optimization [C]// Proc of IEEE International Conference on Neural Networks. 1995: 1942-1948.

[3] 李炳宇，萧蕴诗，吴启迪. 一种基于粒子群算法求解约束优化问题的混合算法 [J]. 控制与决策，2004, 19 (7): 804-807. (Li Bingyu, Xiao Yunshi, Wu Qidi. A hybrid algorithm based on particle swarm algorithm for solving constrained optimization problems [J]. Control and Decision, 2004, 19 (7): 804-807.)

[4] 王俊伟，汪定伟. 粒子群算法中惯性权重的实验与分析 [J]. 系统工程学报，2005, 20 (2): 194-198. (Wang Junwei, Wang Dingwei. Experiment and

analysis of inertia weight in particle swarm optimization [J]. Journal of Systems Engineering, 2005, 20 (2): 194-198.)

[5] 李晓磊. 一种新型的智能优化方法-人工鱼群算法 [D]. 杭州: 浙江大学, 2003. (Li Xiaolei. A new intelligent optimization method-artificial fish school algorithm [D]. Hangzhou: Zhejiang University, 2003.)

[6] Li Xiaolei, Shao Zhijiang, Qian Jixin. An optimizing method based on autonomous animate: fish swarm algorithm [J]. Systems Engineering-Theory & Practice, 2002 (11): 32-38.

[7] 段海滨, 王道波, 朱家强, 等. 蚁群算法理论及应用研究的进展 [J]. 控制与决策, 2004, 19 (12): 1321-1326. (Duan Haibin, Wang Daobo, Zhu Jiaqiang, etal. Advances in theoretical and applied research of ant colony algorithm [J]. Control and Decision, 2004, 19 (12): 1321-1326.)

[8] 吴亮红. 差分进化算法及应用研究 [D]. 长沙: 湖南大学, 2007. (Wu Lianghong. Differential evolution algorithm and its application [D]. Changsha: Hunan University, 2007.)

[9] Yang Chenguang, Tu Xuyan, Chen Jie. Algorithm of marriage in honey bees optimization based on the wolf pack search [C]// Proc of International Conference on Intelligent Pervasive Computing. Washington DC: IEEE Computer Society, 2007: 462-467.

[10] 吴虎胜, 张凤鸣, 吴庐山. 一种新的群体智能算法——狼群算法 [J]. 系统工程与电子技术, 2013, 35 (11): 2430-2438. (Wu Husheng, Zhang Fengming, Wu Lushan. A new swarm intelligence algorithm: wolfland algorithm [J]. Systems Engineering and Electronics, 2013, 35 (11): 2430-2438.)

[11] 刘永兰, 李为民, 吴虎胜, 等. 基于狼群算法的无人机航迹规划 [J]. 系统仿真学报, 2015, 27 (8): 1838-1843. (Liu Yonglan, Li Weimin, Wu Husheng, etal. Unmanned aerial vehicle trajectory planning based on wolf group algorithm [J]. Journal of System Simulation, 2015, 27 (8): 1838-1843.)

[12] 王建群, 贾洋洋, 肖庆元. 狼群算法在水电站水库优化调度中的应用 [J]. 水利水电科技进展, 2015, 35 (3): 1-4. (Wang Jianqun, Jia Yangyang, Xiao Qingyuan. Wolf pack algorithm in the optimal operation of hydropower station reservoirs [J]. Advances in Science and Technology of Water Resources, 2015, 35 (3): 1-4.)

[13] 惠晓滨, 郭庆, 吴娉娉, 等. 一种改进的狼群算法 [J]. 控制与决策, 2017, 32 (7): 1163-1172. (Hui Xiaobin, Guo Qing, Wu Pingping, *et al.* An improved wolf pack algorithm [J]. Control and Decision, 2017, 32 (7): 1163-1172.)

[14] 钱荣鑫. 一种基于文化机制的狼群算法 [J]. 信息技术, 2015 (12): 98-102. (Qian Rongxin. A wolf algorithm based on cultural mechanism [J]. Information Technology, 2015 (12): 98-102.)

[15] 曹爽, 安建成. 改进狼群优化算法的 Otsu 图像分割法 [J]. 微电子学与计算机, 2017, 34 (10): 16-21. (Cao Shuang, An Jiancheng. Otsu image segmentation method for improved wolf group optimization algorithm [J]. Microelectronics & Computer, 2017, 34 (10): 16-21.)

[16] Li Junhua, Li Ming. Convergence analysis and convergence rate estimate of cellular genetic algorithms [J]. Pattern Recognition & Artificial Intelligence, 2012, 25 (5): 874-878.

[17] Galletly J. Evolutionary algorithms in theory and practice [J]. Complexity, 1996, 2 (8): 26-27.

[18] Solano-Aragón C, Castillo O. Optimization of benchmark mathematical functions using the firefly algorithm with dynamic parameters [M]// Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics. [S. l.] : Springer International Publishing, 2015: 81-89.